Redundancy Management Thread
Atlas DP1

Checkout and Launch Control System (CLCS)

84K00303-009

Approval:

Chief, System Engineering and Integration Division   Date

Table of Contents

1. INTRODUCTION 1

1.1 REDUNDANCY MANAGEMENT THREAD OVERVIEW. 1
1.2 REDUNDANCY MANAGEMENT THREAD CONCEPT 1
1.3 OPERATIONAL AND FUNCTIONAL OVERVIEW 2
1.3.1 System Integrity 2
1.3.2 Subsystem Integrity 5
1.1.2 Subsystem Monitoring by Subsystem Integrity 12
1.3.6 System Configuration Table 12
1.3.7 Subsystem States 13
1.3.8 System Event Codes 14
1.4 REDUNDANCY MANAGEMENT SPECIFICATION 16
1.4.1 Statement of Work 16
1.4.2 Requirements 18
1.5 REDUNDANCY MANAGEMENT HARDWARE DIAGRAM 20
1.6 REDUNDANCY MANAGEMENT DELIVERABLES 20
1.7 REDUNDANCY MANAGEMENT ASSESSMENT SUMMARY 21
Labor Assessments 21
Hardware Costs 21
Redundancy Management Procurement 21
1.8 REDUNDANCY MANAGEMENT SCHEDULE & DEPENDENCIES 22
Schedule 22
Dependencies 22
1.9 REDUNDANCY MANAGEMENT SIMULATION REQUIREMENTS 22
1.10 REDUNDANCY MANAGEMENT INTEGRATION AND SYSTEM TEST PLAN 22
1.11 REDUNDANCY MANAGEMENT TRAINING REQUIREMENTS 24
1.12 REDUNDANCY MANAGEMENT FACILITIES REQUIREMENTS 24
1.13 TRAVEL REQUIREMENTS 24
1.14 REDUNDANCY MANAGEMENT ACTION ITEMS/RESOLUTION 24

2. CSCI ASSESSMENTS 24

2.1 SYSTEM ENGINEERING ACTION ITEM 24
2.2 SYSTEM CONTROL ASSESSMENT 24
2.3 COMMAND SUPPORT 27
2.4 SYSTEM VIEWERS 28
2.5 SYSTEM SERVICES ASSESSMENT 29
2.6 DATA DISTRIBUTION ASSESSMENT 30
2.7 LDB GATEWAY ASSESSMENT 33
2.8 PCM DOWNLINK GATEWAY 34
2.9 GSE GATEWAY ASSESSMENT 34
GSE GATEWAY SERVICES ASSESSMENT 34

CSCI Assessment 35

3. HWCI ASSESSMENTS 36

Table of Figures

Figure 1 — System/Subsystem Integrity Topology 2
Figure 2 Redundancy Management Data Flow 3
Figure 3 Subsystem Integrity Data Flow 6
Figure 4 — Generic Subsystem & Subsystem Integrity 7
Figure 5 Data Distribution Redundancy 9
Figure 6 Redundant Constraint Management Concept 11

Table of Tables

Table 1 Faults Indicating GS1A Failure....4
Table 2 — Active/Standby Subsystem Transmission Rate....7
Table 3 System Event Codes....15

Assessment Team

Name	CI Represented	Phone
Bateman, Brian	System Control	1-9075
Blackledge Jack	SE Thread Lead	1-9078
Bullington, Van	LDB GW SSI	1-4654
Castner, Ken	Redundancy Management	1-9020
Duncan, Carl	O&M User	1-6124
Davis, Frank K.	Integration and Test	1-7313
Davis, Steve	Network Services	1-2222
Hrezo, Gary	System Integrity Viewer	1-7445
Jamieson, Tom	Systems Engineering	1-9088
King, Charla	Test Build CSCIs	1-7587
Le, Chau	PCM GW SSI	1-2293
Lunceford, Mike	GSE GW SSI	1-7557
McMahon, Bob	System Services	1-9045
Moore, Steve	Ops CM/Activity Manager	1-9103
Penn, Ronald M.	Integration and Test	1-6124
Raucci, Jack	Data Distribution	1-7493

1. INTRODUCTION

1.1 REDUNDANCY MANAGEMENT THREAD OVERVIEW.

This thread develops the Atlas Redundancy Management using the System and Subsystem Integrity infrastructure developed in Thor. The Thor delivery defined the means for transmitting/logging, and displaying Subsystem health and performance information. For Atlas, subsystem failure indications will be collected and active/standby switchovers will be commanded. The set configurability capability will be extended to include the capability to add and remove subsystems from a test set and update the System Configuration Table (SCT). During Atlas, the mechanisms for supporting application software redundancy will be defined, but not implemented.

1.2 REDUNDANCY MANAGEMENT THREAD CONCEPT

Status Visibility and Redundancy Management in an RTPS Test Set is accomplished by System Integrity, Subsystem Integrity, and System Status Viewer. System Integrity is the focal point for redundancy management policy implementation and health and status data collection. Subsystem Integrity in each of the subsystems is responsible for collecting health and status information for the subsystem and reporting it to System Integrity. Subsystem Integrity also reports to System Integrity when it detects failures in other subsystems (standby subsystems report on active subsystems, receivers of data report on suppliers).

System Integrity executes in the Test Set Master CCP and implements the redundancy management policies based on data collected from each of the subsystems and a set of redundancy management rules. Subsystem Integrity executes in every subsystem in the Test Set and reports subsystem health, status, and activity within the local subsystem to System Integrity. The System Status viewer executes in any Command and Control Workstation and provides the user with the overall status of the Test Set or the detailed status of any subsystem in the Test Set. The user may also execute commands from the System Status Viewer to reconfigure the Test Set by switching between active and standby subsystems or deactiving subsystems. **Figure 1 — System/Subsystem Integrity Topology** below shows the allocation of System and Subsystem Integrity in a typical RTPS Test Set.

CLCS Redundance Management - System & Subsystem Integrity

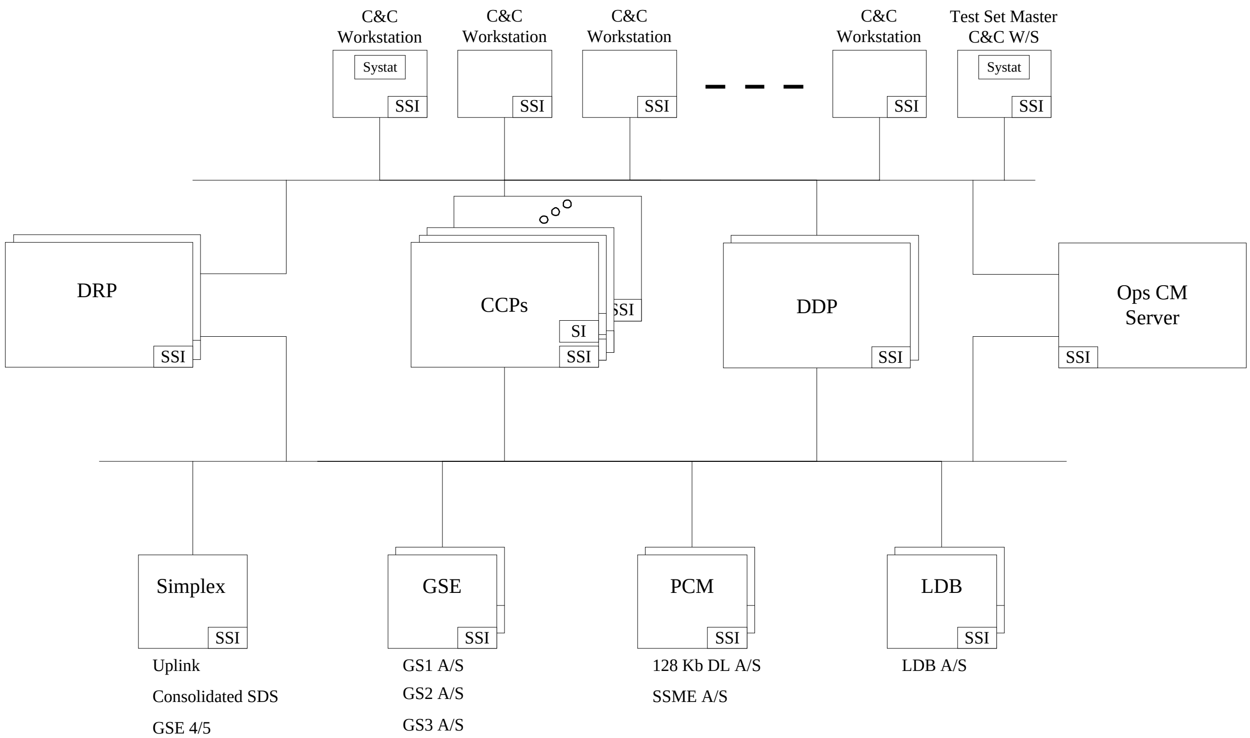


Figure 1 — System/Subsystem Integrity Topology

Note: The Uplink, Consolidated SDS, and GSE 4/5 Gateways are shown here as Simplex systems, all subsystems will be designed to operate as redundant systems. The decision to run a subsystem as a simplex or redundant system is an operational one made by O&M personnel based on test requirements for availability and the availability of hardware to execute redundant subsystems.

1.3 OPERATIONAL AND FUNCTIONAL OVERVIEW

1.3.1 System Integrity

System Integrity is responsible for implementing the redundancy management policies based on information provided by *test set network integrity* and the subsystem integrity CSC's in each subsystem. *System Integrity consists of two parts; the Fault Detection and Identification (FDI) function and the Corrective Action function as illustrated in* ***Figure 1****. System Integrity uses FDI to evaluate the fault and system status information. FDI uses a set of Fault Detection Rules that define failure modes based on the logical combination of events and their probabilities and identify the most probable fault source. The Fault Detection and Identification then forwards the failure mode information to the Corrective Action function. The Corrective Active function then issues the appropriate commands to recover from the failure based on the current operating environment of the Test Set.*

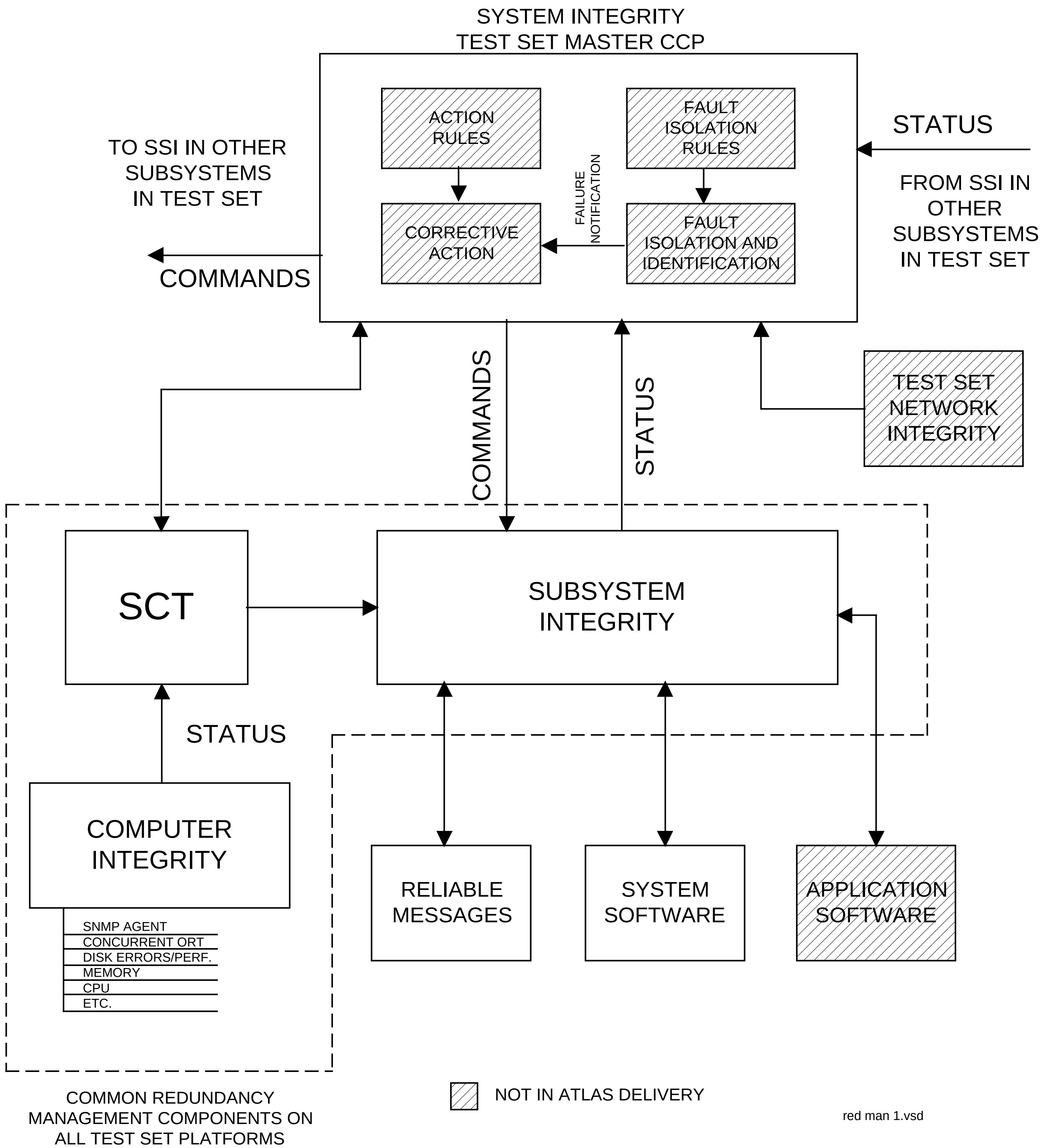


Figure 2 Redundancy Management Data Flow

<u>EXAMPLE:</u>

The following discussion is an example of a simple implementation using a boolean logic engine to evaluate a set of rules that are maintained by system integrity. For Atlas, system integrity will use this, or a similar technique, to implement the redundancy management policy. During Atlas, resources permitting, other techniques will be evaluated.

Table 1 is a truth table that describes the fault inputs that are used to determine that the GS1A gateway has failed. A more complete analysis of gateway failure modes is presented in Appendix A.

	GS1A Reports missing HIM responses SEC 1	GS1S reports missing HIM responses SEC 2	GS1S reports missing GS1A polls SEC 3	DDPA reports missing GS1A packet SEC 4	DDPS reports missing GS1A packet SEC 5	DDPA reports missing GS1S requested packet SEC 6	DDPS reports missing GS1S requested packet SEC 7
1	0	0	0	1	1	0	0
2	0	0	1	1	1	0	0
3	0	1	1	0	0	0	0
4	0	1	1	1	1	0	0
5	1	0	0	0	0	0	0
6	1	0	1	0	0	0	0
7	1	1	1	0	0	0	0
8	1	1	1	1	1	0	0

Table 1 Faults Indicating GS1A Failure

NOTE: This example assumes some fault reporting capabilities that will not be implemented in Atlas.

The conditions indicating a GS1A failure can be represented by the following logic equation written in simple min term format (not reduced making it easy to correlate to the table):

GS1A_Fail = (SEC 4 & SEC 5) + (SEC 3 & SEC 4 & SEC 5) + (SEC 2 & SEC 3) + (SEC 2 & SEC 3 &
SEC 4 & SEC 5) + SEC 1 + (SEC 1 & SEC 3) + (SEC 1 & SEC 2 &
SEC 3) + (SEC 1 & SEC 2 & SEC 3 & SEC 4 & SEC 5)

(& - Logical AND
+ - Logical OR)

Additional information may be obtained from the information presented in the table. For example, from Table 1, row 1, the inputs from the active and standby DDP's indicate that GS1A is unable to provide change data. In addition, the fact that GS1S did not report missing HIM responses or polls, indicates the gateway is still able to communicate with the HIM's but is not able to communicate on the RTCN with the DDP's. This failure mode can be expressed with the equaiton:

GS1A_Fail_NIC = -SEC 1 & -SEC 2 & - SEC 3 & SEC 4 & SEC 5 & -SEC 6 & -SEC 7
(ROW 1FAILURE INDICATION)

The fourth row indicates GS1A has lost the ability to communicate with the HIM's. This failure mode can be expressed with the equation:

GS1A_Fail_HIM_IF = -SEC 1 & SEC 2 & SEC 3 & SEC 4 & SEC 5 & -SEC 6 & -SEC 7

The table and the equation represent the rules that are used to determine that GS1A has failed. This set of rules does not indicate the action to be taken when the failure occurs. The failure indication is forwarded to the Corrective Action function.

The Corrective Action function accepts failure input information and, using its corrective action rules, determines what action. An example of a corrective action for a GS1A failure is:

GS1S_TO_Active = GS1A_Fail & GS1S_GO & GS1_Switch_Enabled

Another rule might be:

Restart_GS1A = GS1A_Fail & (GS1S_NOGO + GS1_Switch_Disabled)

END OF EXAMPLE:

Both Active and Standby System Integrity receive the Subsystem Health FDs for all of the Subsystems in the Test Set and analyze them to determine if any switch-overs are needed. If no changes are needed both Active and Standby System Integrity update their Current Status Information. If a health counter is missed Active System Integrity notes the fact. If the next expected health counter or data packet is received, Active System Integrity outputs a System Message. If two successive Health FD updates or Data Change Packets are missed for a subsystem, Active System Integrity will cause a switch-over to take effect according to the following rules:

1. Subsystem is redundant
2. Standby Subsystem is ready to assume active role
3. Switch-over is enabled for the redundant pair

Standby System Integrity monitors the health of Active System Integrity. If the health counter of the Active System Integrity is missed, Standby System Integrity notes the fact and continues. If the next expected health counter or data packet is received, Standby System Integrity outputs a System Message. If Active System Integrity fails to update its health counter FD for two consecutive times Standby System Integrity will command Active System Integrity to the standby state and become active.

1.3.2 Subsystem Integrity

Data collection and status reporting is accomplished by a set of programs, Subsystem Integrity, that execute in each subsystem of the test set. Subsystem Integrity is responsible for:

1. Monitoring health and status of subsystem software and hardware
2. Communicating subsystem health failures events to System Integrity
3. Communicating status and performance information
4. Maintaining the local copy of the System Configuration Table as directed by System Integrity
5. Providing an interface to the acquire data from the System Configuration Table

Subsystem Integrity in each of the subsystems monitors the health of the hardware, system software, and application software as illustrated in **Table 2**. Subsystem Integrity gathers platform hardware status and error information from Computer Integrity. Computer Integrity acquires status from *a COTS SNMP agent and from Concurrent ORT.* Health counters and operating systems status information is gathered for system and application software. Reliable Messages provides the health and status for the subsystem's interfaces to the attached networks.

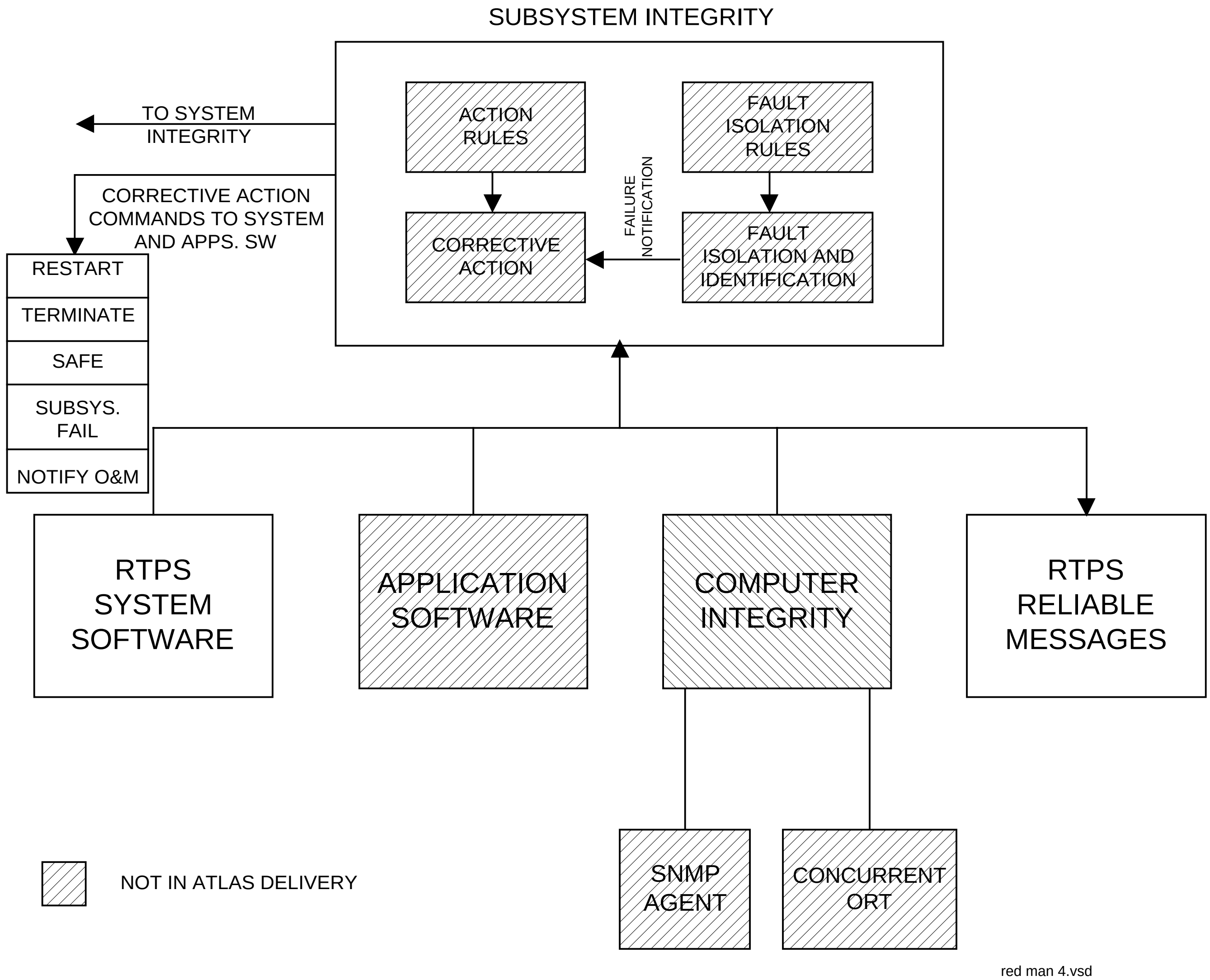


Figure 3 Subsystem Integrity Data Flow

Subsystem Integrity uses a rule based engine to determine failure modes from input fault indications similar to the System Integrity FDI. The corrective action for a system or application software failure on a subsystem is specified by a set of rules on the subsystem. The corrective actions include:

1. *Restart the process*
2. *Terminate the process*
3. *Start a safing process*
4. *Send subsystem failure notice to System Integrity*
5. *Send System Message to O & M*

A set of System Status FDs are defined for all subsystems and subsystem devices. Subsystem Integrity in each of the subsystems creates and maintains the data required by the FDs and introduces these FDs as System Status FDs at the appropriate rate into the data stream as depicted in Figure 4. Data Distribution processes the System Status FDs as pseudo FD's. Table 2 — Active/Standby Subsystem Transmission Rate contains estimates for transmission rates for Subsystem Health Counter and Subsystem Status FDs.

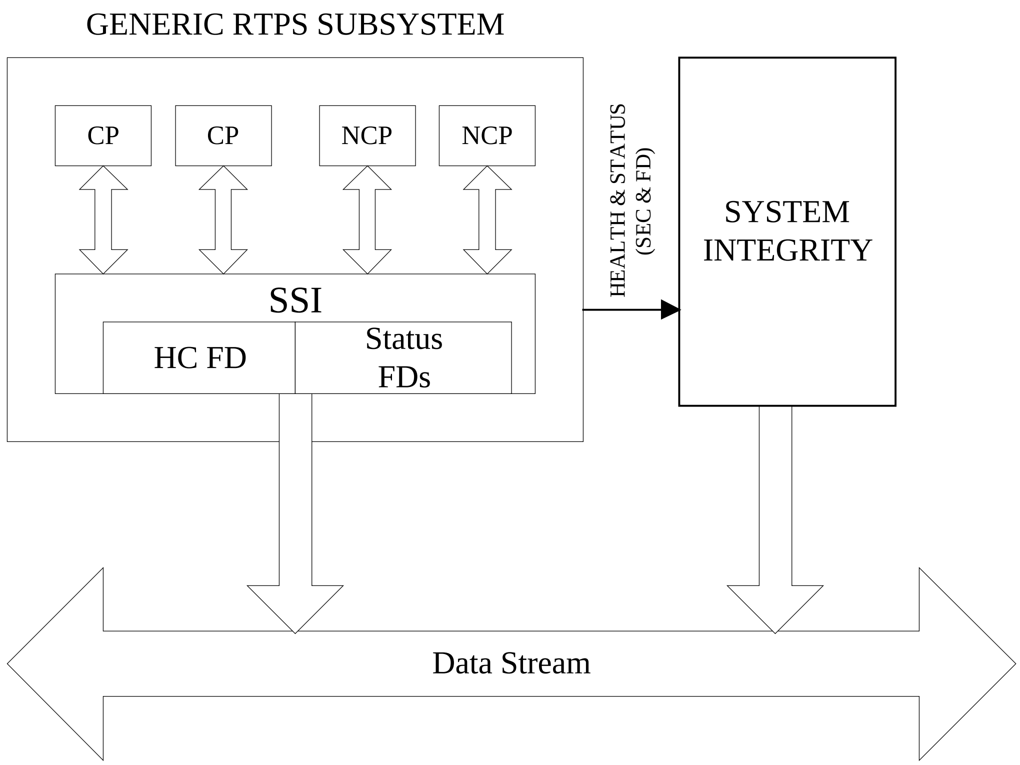


Figure 4 — Generic Subsystem & Subsystem Integrity

Subsystem Integrity in each of the RTPS Subsystems introduces Subsystem Health FDs and Subsystem Status FDs (e.g., Health Counters, Use & Error Counters, etc.) as follows:

	Active Subsys Xmit Rate		Standby Subsys Xmit Rate		Hot Spare	
Subsystem	HC FD	Status FDs	HC FD	Status FDs	HC FD	Status FDs
GSE G/W	SSR	P/C/D	SSR	P/C/D	1/Sec	P/C/D
PCM DL G/W	SSR	P/C/D	SSR	P/C/D	1/Sec	P/C/D
SSME GW	SSR	P/C/D	SSR	P/C/D	1/Sec	P/C/D
LDB G/W	SSR	P/C/D	SSR	P/C/D	1/Sec	P/C/D
PCM UPLK G/W	SSR	P/C/D	SSR	P/C/D	1/Sec	P/C/D
Consolidated G/W	SSR	P/C/D				
DDP	SSR	P/C/D	SSR	P/C/D	1/Sec	P/C/D
CCP’s	SSR	P/C/D	SSR	P/C/D	1/Sec	P/C/D
Ops CM Server	1/Sec	P/C/D				
DRP	1/Sec	P/C/D	1/Sec	P/C/D		
C & C W/S’s	1/Sec	P/C/D				

Table 2 — Active/Standby Subsystem Transmission Rate

P/C/D => P = Periodically — At the rate defined for the FDs. C = Change - When an error occurs in a Subsystem, all System Status (i.e., use, error, and performance) FDs for that Subsystem will be introduced into the data stream at the next update cycle for the HC FD. D = Demand - A capability to update the error counts on demand must also be provided.

1.1.1.1 GSE Gateway Subsystem Integrity

For Atlas, the GSE Gateway will provide support in both the active and standby roles and will support switch-over on command. While in the standby role, the GSE gateway will accept and process all Command and Measurement Descriptor Table update commands. Upon receiving a switch-over command, the standby gateway will perform a switch scan and begin data acquisition. It is anticipated that this process will meet or exceed the fail-over timing requirements.

In the event the above does not satisfy the failover timing requirements, the following procedure will be investigated, post Atlas. A GSE Gateway in the standby role, monitors the response data on the GSE Data Bus to determine if the active Gateway is polling. The standby GSE Gateway collects measurement data, and is prepared to send a backup Change Data packet if requested by Data Distribution. If the Standby Gateway sees no activity on the GSE Data Bus, it notifies System Integrity of "No Bus Activity". GSE Standby Gateway also receives commands from the CCP so that it tracks the Active Gateway and is prepared to issue any commands not issued by the Active Gateway if a switch-over is commanded by System Integrity.

1.1.1.2 LDB Gateway Subsystem Integrity

An LDB Gateway in the standby role, monitors the active LDB Gateway to determine if the active Gateway is still performing its function. During this period the Standby Gateway monitors GPC response data, and is prepared to send a backup response packet if requested. If the Standby Gateway sees no activity on the LDB it notifies System Integrity of "No Bus Activity". LDB Standby Gateway also receives commands from the CCP so that it tracks the Active Gateway and is prepared to issue any commands not issued by the Active Gateway if a switch-over is required and directed by System Integrity.

1.1.1.3 PCM Downlink Gateway Subsystem Integrity

The PCM downlink gateway will provide support in both the active and standby roles. While in the standby role, the PCM D/L gateway will accept and process all Measurement Descriptor Table update commands. The standby PCM D/L gateway will assume the role of the active PCM gateway on command from System Integrity. The standby will activate data acquisition and begin transmitting change data packets after synchronizing with the PCM data stream.

The active and the standby PCM Downlink Gateways perform a checksum on each frame of the input telemetry stream. The checksum and the frame count are transmitted as a system status function designators. A miscompare of the checksum information from the active and standby for the same frame will result in a TBD (system message) to the O & M console.

1.1.1.4 Data Distribution Processor Redundancy Management

The standby DDP shall be capable of replacing the function of a failed active DDP without loss of measurement data or constraint violation notifications, and without sending duplicate constraint violation notifications or measurement changes notifications to applications. From this, there is an implied or derived requirement that the standby DDP has knowledge of the active DDP’s output data streams.

Data Acquisition by Data Distribution Processing (Active and Standby)

Data Distribution in both the active and standby DDP’s receive change data packets from Gateways containing change data and subsystem health FDs. If a change data packet is not received from a Gateway within TBD MSec after it is expected, the data distribution function first requests Reliable Messages to request a retransmission. Reliable Messages sends the retransmission request on both RTCN networks. If the missed change data packet is retransmitted by the active gateway, processing continues. *If the active gateway does not retransmit the missed change data packet, Data Distribution will then send a request to the standby gateway (if it exists) for the change data packet.* Subsystem integrity is notified of the missed packet from the active gateway. Subsystem integrity subsequently notifies System Integrity.

Data Distribution Processing

In RTPS Test Sets with redundant Data Distribution Processor subsystems, the data distribution function in the standby DDP is identical to the Data Distribution function in the active DDP (see Figure 5). The gateway data streams are received by the standby DDP and processed to produce the SSR and DSR change data streams. These data streams are intercepted in the standby DDP prior to Reliable Messages by a new component, the data stream checker. The function of this component is to verify the standby DDP is producing data streams that are equivalent to the data streams from the active DDP (see paragraph below for method of receiving data steams from active DDP). The algorithms for determining equivalence will be determined at design time. These algorithms may include bit-for-bit compare, FDID/time field compares, first and last entry compares, ignoring refresh entries, etc.

The standby DDP executes copies of the CCP and CCWS Data Distribution Receiver components. These components receive the SSR and DSR change data streams, respectively, from the active DDP. As receivers of these data streams, these components must report missing change data packets from the active DDP to subsystem integrity. The SSR change data stream is also used to update a secondary copy of the CVT. The secondary CVT will be compared to the standby generated CVT at predefined intervals (e.g. once per second).

Upon receiving a switch-over command from System Integrity, the standby DDP assumes the role of the active. The data stream check function will complete the check of all queued data from the active DDP and begin transmitting, via Reliable Messages, change data from the local data distribution processing function, picking up at the point the active DDP quit transmitting. The data check function and the data receivers will be deactivated.

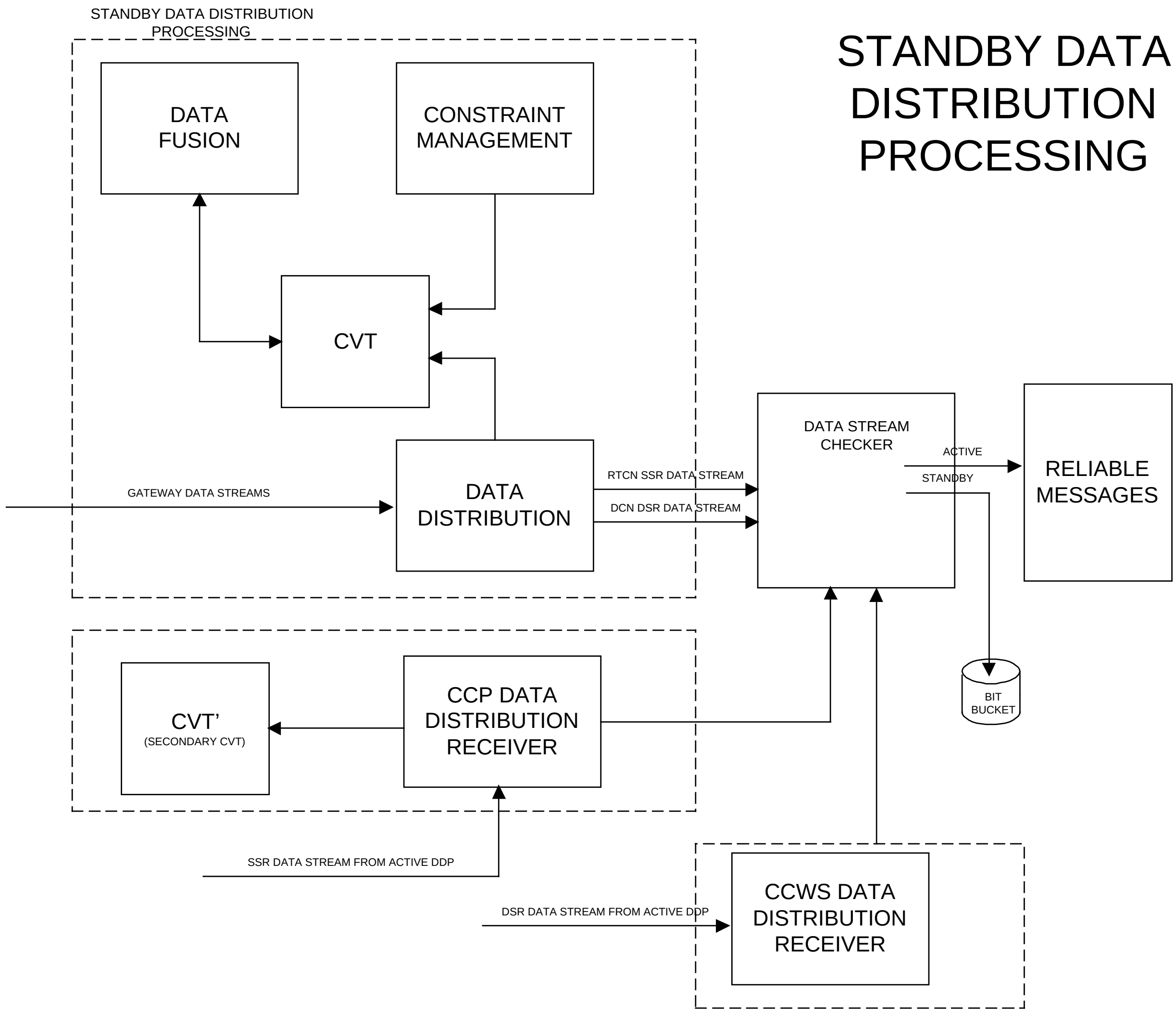


Figure 5 Data Distribution Redundancy

CONSTRAINT MANAGEMENT

In RTPS Test Sets with redundant Data Distribution Processing subsystems, applications in the CCP and CCWS that assert constraints use a common assert function that sends the assert command to both the active and standby DDP. The dual assertion is transparent to the application. A failure to respond from either the active or standby will cause the assert function to notify system integrity. A failure of both the active and the standby DDP to respond will result in an error returned to the asserting process.

Both the active and standby constraint managers receive change data and compute constraint violations. Only the active constraint manager provides constraint violation notification to the application. The standby constraint manager synchronizes its constraint notification output stream with the active, however, the standby will not issue any constraint violation notifications. Any discrepancies between the two streams will cause subsystem integrity to be notified. The standby will keep all constraint notifications pending until it is verified that the active has sent them. Should System Integrity command the standby to become the active, the standby would then issue all pending constraint violation notifications.

An example of a synchronization mechanism is illustrated in Figure 6 Redundant Constraint Management Concept. The active constraint manager notifies applications and the standby constraint manager of constraint violations. The standby constraint manager receives and queues the active constraint violations. The standby constraint manager queues its constraint violation notifications. The two constraint notification queues are compared and all like entries are discarded. If the active fails to compute a constraint violation that matches the standby constraint violation after TBD milliseconds, subsystem integrity is notified. If the active computes a violation not computed by the standby, subsystem integrity is notified.

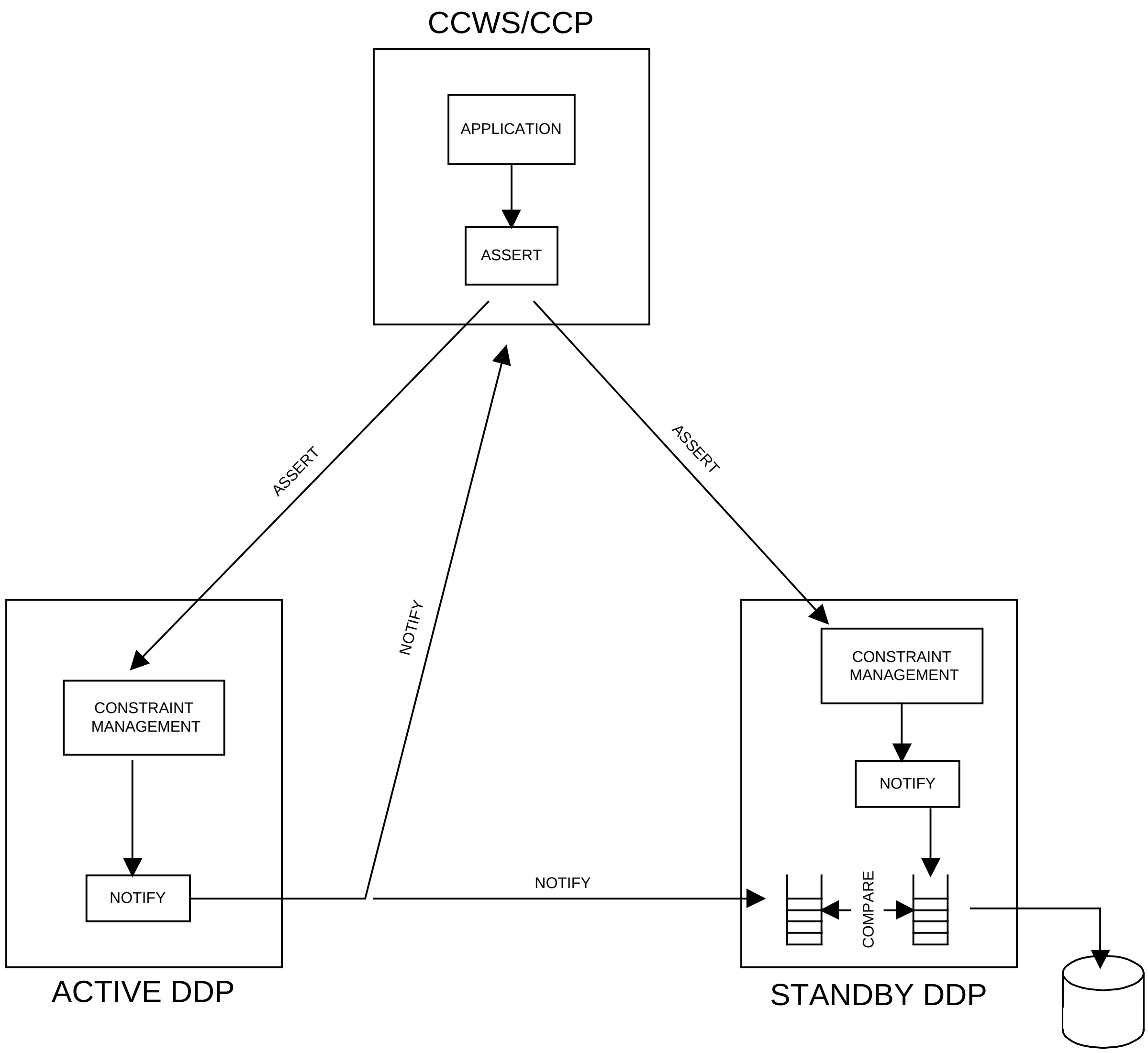


Figure 6 Redundant Constraint Management Concept

Data Fusion

Data Fusion in the standby DDP performs the same data fusion calculations that are being concurrently performed by data fusion in the active DDP. Fusion is performed on data from the standby CVT and included in the standby DDP's output data stream which is checked against the active DDP output.

Data Distribution in the Command and Control Processor

Data Distribution in the CCP receives the change data stream from the active DDP at the System Synchronous Rate (SSR) over the RTCN. If an expected change data packet is not received, a retransmission request is sent to Reliable Messages. Reliable Messages requests a retransmission from the active DDP over both network paths. If the retransmission request fails, subsystem integrity is notified. *If the retransmission fails and a standby DDP is present, the change data packet is requested from the standby DDP.*

Data Distribution in the Command and Control Workstation

Data Distribution in the CCWS receives the change data stream from the active DDP at the Display Synchronous Rate (DSR) over the DCN. If an expected change data packet is not received, a retransmission request is sent to Reliable

Messages. Since the DCN is not redundant, the retransmission request is sent on the same network path that the original transmission would have used. If the retransmission request fails, subsystem integrity is notified. *If the retransmission fails and a standby DDP is present, the change data packet is requested from the standby.*

1.1.2 Subsystem Monitoring by Subsystem Integrity

A set of non-intrusive subsystem checks will be defined for each subsystem and included as part of Subsystem Integrity. These tests will:

1. Analyze the subsystem hardware and software
2. *Detect errors in the hardware and take appropriate action*
   - *Report all critical errors to O&M and System Integrity*
   - *Tally non-critical error and report when they exceed a pre-established error threshold*
3. *Analyze the boot error log and report errors to O&M*
4. Detect errors in the operational status of all essential and non-essential SW processes
5. *Detect and report errors in the software configuration*

These subsystem tests will have the following characteristics:

1. *Run in the background*
2. *Default runtime from subsystem initialization until subsystem termination*
3. *The capability to start and stop by feature and feature group.*

1.1.2.1 Network Monitoring

Subsystem Integrity will monitor the performance and health status of the networks attached to each subsystem. Reliable Messages will notify subsystem integrity of all errors and failed transmissions. In subsystems with connections to both the DCN and the RTCN (CCP's and DDP's), subsystem integrity uses the RTCN to communicate with System Integrity. In the event that all RTCN communications are unsuccessful, subsystem integrity in a DDP or CCP will attempt to communicate with System Integrity via the DCN.

1.1.2.2 Software Monitoring

System software and *application software* are defined as critical or non-critical. Subsystem Integrity monitors the status and Health Count of all critical software functions. (See Appendix C - Health Counters.) If the health of a critical software function is "no go", Subsystem Integrity uses the supplied set of rules to determine what action to take. (See Appendix C for discussion of software health counters.)

COTS software packages that perform non intrusive monitoring of hardware and software functions will be evaluated during Atlas. The evaluation criteria will include:

a) Required performance, status parameters reported?
b) Overhead of package acceptable
c) Does the package accept input to select, control its operations?

1.3.6 System Configuration Table

The System Configuration Table includes the following information for each subsystem assigned to an RTPS Test Set:

1. Subsystem Name
2. Logical Identifier
3. Host Name
4. Role
5. Subsystem Type
6. Physical Identifier
7. Reference Designator
8. Active Primary RTCN IP Address
9. Active Backup RTCN IP Address
10. Standby Primary RTCN IP Address

11. Standby Backup RTCN IP Address
12. Primary DCN IP Address
13. Backup DCN IP Address
14. Switch-over Enabled
15. Current State

The SCT includes the following information for each system software process:

1. Process Name
2. Critical/Non critical
3. Periodic/Non-periodic
4. Frequency
5. Switch-over synchronization method
6. Health counter
7. Restart Indicator

The SCT includes the following information for each application software process:

1. Process Name
2. Critical/Non critical
3. Periodic/Non-periodic
4. Frequency
5. Safing
6. Health counter
7. Restart Indicator

Applications and System Services APIs are provided to access the data in the SCT.
Reliable messages will be notified each time a logical subsystem identifier is changed and a new set of IP addresses are required to address the new subsystem platform.

1.3.7 Subsystem States

A set of Subsystem states are defined as listed below. Any subsystem on the Test Set can exist in one of these states. Appendix D contains the current design of the Subsystem State Matrix.

Subsystem not in Configuration
The subsystem has been assigned to the test set, however, it has not been assigned a role in the test activity. The subsystem hardware is available to be included in the activity without additional manual intervention (patching). A control group or CCWS subsystem may have its operating system running (platform initialized), however it cannot perform SCID/TCID initialization. A gateway subsystem may have its operating system and SCID initialized,

Subsystem in Configuration

The subsystem has been assigned a role as an active, standby, or hot spare in a test set. The subsystem platform can be initialized. The subsystem can accept a SCID/TCID initialize command.

Platform Not Initialized

The Platform Not Initialized mode is the mode in which a platform either has no operating system installed or has an operating system installed but is powered off or any other condition which would not qualify it to be in one of the following modes.

Platform Initialized

The Platform Initialized mode is achieved when the operating system (Unix, VxWorks.) for the subsystem has been loaded and initialized. The Unix configuration and environment parameters are established by a set of scripts that tailor the operating system to the RTPS configuration. An Operations Configuration management daemon is initialized during platform load. Network services, including Reliable Messaging, are also initialized. The NTP is initialized and GMT is available. For the gateways, this state includes the SCID. At this point, Pre-Load Diagnostics are run and the results are transferred later to System Integrity when requested.

SCID/TCID Loaded

The SCID/TCID Loaded mode indicates that all of the software required for the subsystem to perform its intended function in the RTPS Test Set to which it has been assigned has been loaded on the subsystem hard disk. A load verification test is performed and the results forwarded to System Integrity.

Communicating

The Communicating mode is achieved when a subsystem platform has initialized the software components that are necessary to send RTPS packet payloads to other subsystems via Reliable Messages. The platform Subsystem Integrity begins sending a health counter to System Integrity.

Ready

The Ready mode is achieved after all RTPS systems software processes are spawned. In the Ready mode, all of the RTPS system software functions for that subsystem that are necessary for supporting test operations are initialized and executing. Only configuration and initialization commands are recognized by the subsystem in this state. Redundancy management active/standby synchronization begins with this state. Gateways can accept MDT Maintenance Commands in this mode but not end item commands.

Go

The Go mode is achieved when the executable applications software has been loaded and initialized. At this point, the subsystem is fully operational and can accept Test specific commands as well as configuration commands. Configuration commands will include a command to regress to a previous state.

1.3.8 System Event Codes

SEC	Name	Source	Destination
1-255	HIM Status Change		
SEC256	Subsystem Loaded	SSI	SI (Master SCT)
SEC257	Subsystem Comm.	SSI	SI (Master SCT)
SEC258	Subsystem Go	SSI	SI (Master SCT)
SEC259	Subsystem NoGo	SSI	SI (Master SCT)
SEC260	Subsystem Not Comm.	SSI	SI (Master SCT)
SEC261	Subsystem Not Loaded	SSI	SI (Master SCT)
SEC262	Terminate	SI	SSI on trgt pltfrm
SEC263	Switchover Directive	SI	SSI on trgt pltfrm
SEC264	New Active	SI	All SSI (Local SCT)
265-328	PCM/UPLK,UCS,LDBA status change for 1 of up to 64 FDs. Bits 15-0 = System Event Code (identifies 1 to 64 FDs).		
329-392	PCM area format Id for 1 of up to 64 areas. Bits 15-0 = System Event Code (identifies 1 to 64 areas).		
SEC393	Subsystem ORT	SSI	SI (Master SCT)
SEC394	Subsystem Not ORT	SSI	SI (Master SCT)
SEC395	No Pkt rcvd frm gtwy	SI-DDP	SI-CCP
SEC396	Stdby GSE detctd no poll frm actve GSE	GSEnS	SI

SEC397	GSE rpts no rspnse frm bus	GSEnA	SI
SEC398	HC not Incremented	SI-DDP	SI-CCP
SEC399	HC has Decremented	SI-DDP	SI-CCP
SEC400.	Terminate Gracefully	SI	SSI on trgt pltfrm
SEC401	Subsystem Loaded	SI (Master SCT)	All SSI (Local SCT)
SEC402	Subsystem Comm.	SI (Master SCT)	All SSI (Local SCT)
SEC403	Subsystem Go	SI (Master SCT)	All SSI (Local SCT)
SEC404	Subsystem NoGo	SI (Master SCT)	All SSI (Local SCT)
SEC405	Subsystem Not Comm.	SI (Master SCT)	All SSI (Local SCT)
SEC406	Subsystem Not Loaded	SI (Master SCT)	All SSI (Local SCT)
SEC407	Subsystem In Config	SI (Master SCT)	All SSI (Local SCT)
SEC408	CPU Utilization	CI	SI (Master SCT)
SEC409	Available Memory	CI	SI (Master SCT)
SEC410	Disk Utilization	CI	SI (Master SCT)
SEC411	Disk Access	CI	SI (Master SCT)
SEC412	Disk Errors	CI	SI (Master SCT)
SEC413	Initial HC Received	SI-DDP	SI (Master SCT)
SEC414	Subsystem Role	SSI	SI (Master SCT)
SEC415	Subsystem Swtchovr En	SSI	SI (Master SCT)
SEC416	Subsystem Exectng On	SSI	SI (Master SCT)
SEC417	Resource IP Address	SSI	SI (Master SCT)
SEC418	Resource Ref Des	SSI	SI (Master SCT)
SEC419	Resource Physical Id	SSI	SI (Master SCT)
SEC420	Resource Host Name	SSI	SI (Master SCT)
SEC421	Resource Executing	SSI	SI (Master SCT)
SEC422	Resource Phys Name	SSI	SI (Master SCT)
SEC423	Computer Ser Num	SSI	SI (Master SCT)
SEC424	Subsystem Role	SI (Master SCT)	All SSI (Local SCT)
SEC425	Subsystem Swtchovr En	SI (Master SCT)	All SSI (Local SCT)
SEC426	Subsystem Exectng On	SI(Master SCT)	All SSI (Local SCT)
SEC427	Resource IP Address	SI (Master SCT)	All SSI (Local SCT)
SEC428	Resource Ref Des	SI (Master SCT)	All SSI (Local SCT)
SEC429	Resource Physical Id	SI (Master SCT)	All SSI (Local SCT)
SEC430	Resource Host Name	SI (Master SCT)	All SSI (Local SCT)
SEC431	Resource Executing	SI (Master SCT)	All SSI (Local SCT)
SEC432	Resource Phys Name	SI (Master SCT)	All SSI (Local SCT)
SEC433	Computer Ser Num	SI (Master SCT)	All SSI (Local SCT)
SEC434	Subsystem ORT	SI (Master SCT)	All SSI (Local SCT)
SEC435	Subsystem Not ORT	SI (Master SCT)	All SSI (Local SCT)
436-499 Avail.			
SEC500	SCT Relinquish Rqst	SI (Master SCT)	SI (Acting Master)
SEC501	SCT Relinquish Rspnse	SI (Master SCT)	SI(Acting Master)
SEC502	/SCT Relinquish Accpt	SI (Master SCT)	SI (Acting Master)
SEC503	SCT Master Request	SSI(Local SCT)	SI (Master SCT)
SEC504	SCT Master SCT Assert	SSI(Local SCT)	All SSI (Local SCT)
SEC505	SCT Master SCT Ack	SI (Master SCT)	SSI (Local SCT)
SEC506	SCT Update ReRequest	SSI(Local SCT)	SI (Master SCT)
SEC507	SCT Master Req Resp)	SI (Master SCT	SI (Acting Master)
SEC508	SCT Updt ReReq Resp	SI (Master SCT)	SI (Acting Master)
509-64K Avail.			

Table 3 System Event Codes

1.4 REDUNDANCY MANAGEMENT SPECIFICATION

1.4.1 Statement of Work

Analyze the SLS and "Other Requirements" that are included and provide an assessment in DP1 of:

- Whether the requirement is incorporated into the Atlas release,
- The level of maturity the implementation will achieve in Atlas
  - Low = function only implemented in one subsystem,
  - Medium = function implemented in multiple CSCIs/Subsystems, but capability not available across the entire system,
  - High = function is implemented nearly everywhere, or
  - Complete = function is implemented everywhere that it is needed
- If the requirement will have to be verified for HMF to be declared operational

General

- From Real-Time Critical Network based subsystems, provide Subsystem Integrity information to System Integrity at the System Synchronous Rate. [Complete]
- Report Required Subsystem integrity data to System integrity. [Complete]
- Provide collection and detection of system failures.[Medium]
- Provide notification of failures.
  - System Messages [High]
  - Event Notification to register applications [High]
  - System status FDs [High]
- Provide Management from System Integrity to maintain system status.
  - Update FD status on System unavailability. [Complete]
  - Update Subsystem Status based on state. [Complete]
- Provide coordination of subsystem redundancy switch-over.
  - Enable and disable Subsystem as active [Complete]
  - Enable and disable subsystem as standby. [Complete]
  - Enable and disable subsystem for failover. [Complete]
  - Direct subsystem to failover. [Medium]
- Update requirement Matrix in SLS

GSE Gateway

- Provide detection and reporting of:
  - Loss of Ground Data Bus
  - Loss of Hardware Interface Module
  - Loss of Hardware Interface Module Card
  - Detectable Subsystem Failures.
  - Failure of Active/Standby pair.

- Execute the following actions:
  - Re-initialize Hardware Interface Module input scans.
  - Re-initialize Hardware Interface Module output states.
  - Completing incomplete Command transactions.
  - Switch Real-Time Critical Network Networks [Complete]

LDB Gateway

- Provide detection and reporting of:
  - Loss of Launch Data Bus
  - Detectable Subsystem Failures.
  - Failure of Active/Standby pair.
- Execute the following actions:
  - Switch Launch Data Bus
  - *Complete incomplete Command transactions (TITAN)*
  - *Synchronizing transactions between Active/Standby pair (TITAN)*
  - Switch Real-Time Critical Network Networks [Complete]

OFI PCM and *SSME Gateway*

- Provide detection and reporting of:
  - Loss of Down Link Signal (Signal Level Bit Sync, Frame Sync)
  - Detectable Subsystem Failures.
  - Failure of Active/Standby pair.
- Execute the following actions:
  - Switch Down link inputs
  - Switch Real-Time Critical Network Networks [Complete]

Data Distribution Processing

- Provide capability to perform true redundant processing
- Provide detection and reporting of:
  - Loss of packet data from source provider [Complete]
  - Detectable Subsystem Failures. [Partial]
  - Failure of Active/Standby pair. [Partial]
- Execute the following actions:
  - Source data selection [Complete]
  - Data invalidation on provider failure. [Low]
  - Re initialize on failover of Data Processing [Partial]
  - Synchronizing of data table between Active/Standby pair [Complete]
  - Switch Real-Time Critical Network Networks [Complete]

Command and Control Processing

- Provide detection and reporting of:
  - Loss of packet data from source provider [Complete]
  - Detectable Subsystem Failures. [Partial]
  - *Failure of Application Software*
- Execute the following actions: [Partial]
  - *Provide method to direct Applications or Application sets to a safe state.*

    - *Provide method to direct Applications or Application sets to stop processing*
    - *Provide method to direct Applications or Application sets to start processing*
    - *Provide method to direct Applications or Application sets to a switch over state processing*
    - Switch Real-Time Critical Network Networks [Complete]

Command and Control Workstation

- Provide detection and reporting of:
    - Loss of packet data from source provider [Complete]
    - Detectable Subsystem Failures. [Partial]
    - Failure of Display Software
- Execute the following actions:
    - *Provide method to direct Applications or Application sets to stop processing*
    - *Provide method to direct Applications or Application sets to start processing*

Real-Time Critical Network

    - Provide as part of reliable messages a fault tolerant network.

1.4.2 Requirements

SLS Requirements

(SLS 2.1.1.2.12) The RTPS shall provide the capability to perform continuous fault detection and isolation of RTPS subsystem's hardware. [High]

(SLS 2.1.1.2.13) *The RTPS shall provide a set of non-intrusive test programs to test interfaces and subsystem LRUs in the RTPS. [Low]*

(SLS - 2.2.9.1.5) The RTPS shall provide a set of visual displays that provide comprehensive insight into the state and configuration of the set resources (e.g., *network resources*, subsystem assignments, software configuration, etc.). [High]

(SLS - 2.2.9.1.6) The RTPS shall provide different views of Test Sets and activities in configurable sets (e.g., *Master Set View*, Test Set View, Activity View). [High]

(SLS - 2.2.9.2.8) The RTPS shall provide a visual display depicting the health and status of all hardware resources within a Test Set and *within all Test Sets of a Configurable Set*. [High]

(SLS - 2.2.9.2.9) The RTPS shall provide the capability to monitor the configuration of each subsystem participating in a test including what software is executing and any subsystem error conditions.[Medium]

(SLS - 2.2.9.2.10) The RTPS shall provide a central point for the display of system error, status, and mode change messages. [High]

(SLS - 2.2.9.2.15) The RTPS shall provide the capability to continuously monitor subsystem resource utilization in all RTPS subsystems. [Medium]

(SLS - 2.2.9.3.1) The RTPS shall provide redundancy management of all redundant subsystems and network resources in a Test Set. [Medium]

(SLS - 2.2.9.3.2) The RTPS shall provide a central point to coordinate and direct redundant element activation (known as System Integrity). [Complete]

(SLS - 2.2.9.3.3) System Integrity shall be capable of being run from any Console Position within a Test Set. [Complete]

(SLS - 2.2.9.3.4) The RTPS shall provide the capability for a Standby copy of System Integrity to run and monitor the activities of the Active copy of System Integrity. (Titan) [Reference]

(SLS - 2.2.9.3.5) When the Standby copy of System Integrity determines that the Active copy is not operating properly it shall assume the role and responsibilities of the Active System Integrity. (Titan) [Reference]

(SLS - 2.2.9.3.6) The RTPS shall provide a method to share current configuration data with a redundant element. [Complete]

(SLS - 2.2.9.3.7) The RTPS shall provide a method to track redundant element states. [Complete]

(SLS - 2.2.9.3.8) System Integrity shall monitor critical subsystems for failure and in the event a monitored subsystem fails, shall perform a switchover (if enabled) to the standby subsystem. [Medium]

(SLS - 2.2.9.3.9) System Integrity shall report all subsystem errors to a central point. [Complete]

(SLS - 2.2.9.3.10) The CLCS shall provide a reduced capability mode in which a Test Set continues to support even though all copies of System Integrity fail. (Titan) [Partial]

Note: Functions that are not supported or whose capability is reduced in the reduced capability mode are:

1. Redundant Element Switchover
2. Test Set Resource Monitoring
3. Checkpointing
4. Restarting subsystems

(SLS - 2.2.9.3.12) The CLCS shall provide a "warm boot" capability in which System Integrity can be restored after failure.

(SLS - 2.2.9.3.13) After a "warm boot" System Integrity shall restore normal function to those capabilities which were reduced while in the reduced capability mode.

- GSE and PCM Gateways, configured as a redundant pair, shall switch to the standby Gateway with no loss of measurement data and within 1 System Synchronous Rate Time Period of detection. [Complete] [Review]
- For Gateways (except LDB) configured as a redundant pair, switch-over for commands shall be completed in less than 20 milliseconds without any loss of commands. [Partial]
- LDB Gateway switch-over shall be accomplished without any loss of data or commands and shall be completed in less than 500 milliseconds. [Partial]
- The RTPS shall be designed to be Fail Safe. [Partial]
- The RTPS shall be fault tolerant. Specifically, the system shall provide the capability to recover from subsystem failures in the following areas:
  - Command and Control Processing [Partial]
  - Data Distribution Processing [Partial]
  - Critical Data Acquisition Gateways (i.e., LDB, 128 & 192 Kb PCM, GSE) [Partial]
  - Real Time Critical Network and the Display and Control Network [Complete]
- The CLCS shall be designed to have a high level of data integrity. Specifically the system shall provide the following:
  - No loss of command data within the CLCS [Partial]
  - No loss of measurement data within the CLCS [Partial]

- No loss of measurement samples to applications requesting such service [Partial]
- No data which has been corrupted within the CLCS [Partial]
- Health data on a measurement basis [Partial]

- The RTPS shall provide fault tolerance in the Command and Control HCI positions. [Complete]
- The loss of any RTPS Real Time Network component shall not cause switch-over of more than one standby subsystem [Complete]

Requirements Moved to System Control Thread

- The RTPS shall provide the capability to load and initialize the following Software in each subsystem of the Test Set:
  - Platform load [Complete]
  - Subsystem Load (SCID) [Complete]
  - Test SW Load (TCID) [Complete]
- The RTPS shall provide the O&M operator with the capability to select, load, monitor load progress, verify the load, and initialize all Software required in the [Complete]

Other System Requirements

- 4.3.1.6 The system shall provide a method to activate or inhibit active/standby switch over for any redundant subsystem. [Complete]
- 4.3.2.3 The system shall provide a method to read the commanding status of any subsystem. [Complete]
- 4.3.2.6 The system shall provide a method for reading the summary error indicators and counts from any subsystem. [Complete]

Derived Requirements:

Command Management routes System Control Commands (Activate, Switch-over, Terminate, etc.) to System Integrity.

System Integrity receives System Control Commands, performs prerequisite checks, updates the SCT and issues command(s) to subsystem integrity to transition the subsystem to commanded state.

Subsystem Integrity provides the status information and control functions that allow a CCP or DDP to assume active or standby role. This includes inhibiting or allowing output of commands to the network.

System Integrity provides the capability to use a set of rules to implement the failure decision process and the corrective action response to failures.

The standby DDP shall be capable of replacing the function of a failed active DDP without loss of measurement data or constraint violation notifications, and without sending duplicate constraint violation notifications or measurement changes notifications to applications. The standby DDP, therefore, has knowledge of the active DDP's output data streams.

1.5 REDUNDANCY MANAGEMENT HARDWARE DIAGRAM

Not Applicable

1.6 REDUNDANCY MANAGEMENT DELIVERABLES

Deliverable	R&D Document	Code	API Manual	Users Guide
System Integrity	U	U		U
Subsystem Integrity	U	U		
System Status Viewer	U	U		U
Reliable Messages	U	U	U	

Interface Description Document:

IDD Names	Responsible CI	Supporting CI
HWCI to HWCI Name		
CSCI to CSCI Name		
CSC to CSC Name		

Other Example:

COTS Evaluation Trade Study for SNMP tool, rule based decision engine.

1.7 REDUNDANCY MANAGEMENT ASSESSMENT SUMMARY

This section contains the summary of the costs and labor involved in implementing the Atlas Redundancy Management Thread.

Labor Assessments

No.	CSCI/HWCI Name	Atlas LM	Changes covered in
1	System Control	18.5	Redundancy Management Thread
2	System Services	TBD	Redundancy Management Thread
3	Command Support	9.75	Redundancy Management Thread
4	Data Distribution	TBD	Redundancy Management Thread
4	GSE	5.5	Redundancy Management Thread
5	LDB		LDB Gateway Thread
6	PCM		PCM Gateway Thread
7	System Viewers	8	Redundancy Management Thread
8	System Engineering Action	3	Redundancy Management Thread
	TOTAL	44.75 LM	

Hardware Costs

Not Applicable

Redundancy Management Procurement

This section contains a list and schedule of Procurement activities that must be accomplished for the Atlas Redundancy Management thread.

Procurement Activity	**Completion Date**

1.8 REDUNDANCY MANAGEMENT SCHEDULE & DEPENDENCIES

Schedule

Task Name	Start	Finish
Atlas Assessment Kickoff		1/20/98
Concept Panel Internal Review		2/18/98
Concept Panel		2/20/98
Atlas Development		
Requirement Panel Internal Review		4/14/98
Requirement Panel		4/16/98
Design Panel Internal Review		5/12/98
Design Panel		5/14/98
CSCI Code and Unit Testing	6/16/98	7/15/98
CSCI Integration Test	8/3/98	8/21/98
Atlas Development Complete		9/26/98

Dependencies

This section lists dependencies that the Redundancy Management thread has in order to be satisfactorily specified, designed, implemented, or tested.

No.	Dependency Area	Dependency	Need Date
1			
2			
3			
4			

1.9 REDUNDANCY MANAGEMENT SIMULATION REQUIREMENTS

None.

1.10 REDUNDANCY MANAGEMENT INTEGRATION AND SYSTEM TEST PLAN

This section contains the initial plan for CSCI Integration Test (i.e., CIT) and System Level Testing. This plan describes how the capability will be tested both during the CIT and System Test phases.

CIT Test

TCID Required: The validation TCID will be used for Redundancy Management CIT.

System Resources Required: IDE will be used for CIT.
* A second spare CCP/DDP will be required to test CCP and DDP switch-over.

The equipment necessary for the Atlas redundancy management includes:
1. 2 - CCP's (active and standby
2. CCWS
3. 2 - DDP's (active and standby)
4. 2 - GSE Gateways (active and standby)
5. 2 - PCM Gateways (active and Standby)
6. 2 - LDB Gateways
7. OPS CM Server

CSCI's required: System Services, System Control, Application Services (Basic), Data Distribution and Processing, System Viewers (Sysstat, FD Support, System Message Viewer), Command Support, System Control, and GSE, LDB, PCM D/L Gateways.

Additional Data Requirements: System Configuration Table, Failure Detection Rules, Corrective Action Rules.

Test tools: TBD

Test plan:

1. The SCT will be modified by Activity Management to reflect various configurations of the Test Sets.
   - SDE 1 & 2
   - IDE 1
2. Each Test Set will be loaded by Ops CM/Activity Management using the configurations defined above.
3. As the Test Set is loaded the System Status Viewer will monitor the activities to demonstrate the ability to load the system correctly and that the System Status Viewer can track the various states of each of the subsystems in the Test Set.
4. The System Status Viewer will be cycled through the detailed status of each of the Subsystems to demonstrate the availability and correct display of Subsystem Health Counter and Subsystem Status FD Information.
5. Each Subsystem will be forced to fail one by one while viewing the System Status Viewer to demonstrate that all elements of the thread track the failure of the Subsystems. This test will be performed with each system in simplex mode.
6. Bring new subsystems in to replace failed subsystems while viewing with the System Status Viewer to demonstrate that all elements of the thread track the introduction of new Subsystems to replace failed Subsystems.
7. Each Subsystem will be forced to fail one by one while viewing the System Status Viewer to demonstrate that all elements of the thread track the failure of the Subsystems. This test will be performed with each system in redundant mode.
8. Determine if System Integrity has commanded the active system to terminate and the standby subsystem to switch its role to active. Review data recording to determine if change data or commands have been lost
9. Inhibit one network transmission capability on each subsystem. Determine that network has switched and no data transmissions were lost.
10. View network status on System Status Viewer for network failures.

System Test

TCID Required: The validation TCID will be used to test the Redundancy Management CSCIs during System Test.

System Resources Required: IDE will be used for CIT. A second spare CCP/DDP will be required to test CCP and DDP switch-over.

CSCI's required: System Services, System Control, Application Services (Basic), Data Distribution and Processing, System Viewers (Sysstat, FD Support, System Message Viewer), Command Support, System Control, and GSE, LDB, PCM D/L Gateways.

Additional Data Requirements: System Configuration Table, Failure Detection Rules, Corrective Action Rules.

Test tools: TBD

Test plan:

TBD

1.11 REDUNDANCY MANAGEMENT TRAINING REQUIREMENTS

None.

1.12 REDUNDANCY MANAGEMENT FACILITIES REQUIREMENTS

None.

1.13 TRAVEL REQUIREMENTS

None.

1.14 REDUNDANCY MANAGEMENT ACTION ITEMS/RESOLUTION

An additional DDP is needed in the JSC SDE.

2. CSCI ASSESSMENTS

2.1 SYSTEM ENGINEERING ACTION ITEM

1) *Determine the requirements for application software redundancy.*
2) Identify and document the hardware, software, and network failure modes and determine the probability and effect of each failure mode.
3) Review switch-over timing requirements.
4) Refine definitions in Appendix A and update glossary.

2.2 SYSTEM CONTROL ASSESSMENT

OPS CM Manager Work Required

1. The Subsystem Load and Initialization (SLAI) function of OPS CM Manager must coordinate load and initialization states with System Integrity. Prior to each state transition, SLAI must notify System Integrity that the subsystem state will change.
2. Resolve duplicate functions with System Integrity (e.g. application activity count, Kill command)

CSCI Assessment

The labor costs in the table below are for the labor to produce the product in Atlas.

CSC Name	CSC Labor (LM)	% of CSC
OPS CM Manager	TBD	

Basis of estimate

TBD

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation	U	
Users Guide	U	
API Interface Document		
Interface Design Document		
Test Procedure	U	

Assumptions

None.

Open Issues

Provide a list of open issues if there are any. If there are none state none.

COTS Product Dependency List

None.

System Integrity Work Required

System Integrity tasks include:

1. *Evaluate and select a tool to perform a rule based analysis of faults to identify subsystem failures.*
2. Accept System Control command inputs from Command Management and route to appropriate subsystems, software processes.
3. *Provide the subsystem termination control function.*
4. Failure Determination Engine: Implement a data driven engine to determine a failure based on a set of detected symptoms.
5. Failure Isolation/Recovery Engine: use same basic engine as that used for Failure Determination. Will used different primitives.
6. Accept all system/subsystem configuration commands, verify the command is valid given current state, and send the command to the affected computers. "Validation rules" should be data rather than code. Not clear whether these can be incorporated into the Recovery Engine rule set.
7. *Assist system viewer in COTS trade study to determine whether application manager s/w should be used to replace SSI/SI/SysStat viewers. System Viewers to lead effort, SI to contribute to analysis and evaluation (Unicenter, Tivoli, Patrol)*
8. *Redundant Master CCP Operation (SI Switchover) Assume transfer of resulting state data upon completion of command. Most of this already exists through SCT maintenance. Only additional information is the command completed ID.*
9. Extend API as necessary for additional data maintained in the SCT.
10. Add ready state and change go state for CCPs
11. Additional data to be used in Health Determination (RM data, Commanding data...). Actual impact depends on specifics still TBD.

Subsystem Integrity Work Required

Subsystem Integrity Tasks include:

1. Add interfaces to accept configuration commands and trigger user and system applications. Some of these are significant: Termination/Switchover require interactions with applications on the box.
2. Additional data to beincluded in the Subsystem Health calculation or provided to SI. Includes RM information, commanding data. Assume same engine/subset as is used for SI failure determination. The rules/data used will be unique to SSI.
3. Potentially application specific data to be included in subsystem health calculation (assume none in Atlas)

Computer Integrity Work Required

1. Final implementation based on platform selection. Could be entirely COTS based and is tied ito System integrity trade studies.

Network Integrity Work Required

1. Initial development, includes determining necessary data to collect and communication of collected data to SI. Note: Related to Network Management trades study.

SCT Work Required

1. Probably some significant expansion to data collected and maintained. Assume no change in synchronization techniques from Thor
2. Minor changes to table build to support any SCT extensions

CSCI Assessment

The labor costs in the table below are for the labor to produce the product in Atlas.

CSC Name	CSC Labor (LM)	% of CSC
System Integrity	18.5	

Basis of estimate

TBD

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation	U	
Users Guide	U	
API Interface Document		
Interface Design Document		
Test Procedure	U	

Assumptions

None.

Open Issues

Provide a list of open issues if there are any. If there are none state none.

COTS Product Dependency List

None.

2.3 COMMAND SUPPORT

Command Management must notify System Integrity whenever it is sending a command to a subsystem to activate or terminate the subsystem. It must also provide input to system integrity whenever commands are not delivered to their destinations.

Command Support Work Required

Command support must:
1. Route System Control Commands to System Integrity
2. Provide input to Subsystem Integrity when commands time out or are rejected.
3. Add commands to control (activate, terminate, switch over, etc.) CCP's, DDP's, CCWS's.
4. Add Health Counter
5. Notify SSI when commands rejected, time out
6. Determine if destination (subsystem e.g. CCP, G/W, DDP) is in GO mode before issuing end item commands
7. Authenticate System Control commands sent to System Integrity

CSCI Assessment

The labor costs in the table below are for the labor to produce the product in Atlas.

CSC Name	CSC Labor (LM)	% of CSC
Command Manager / Command Interface	6.75	
Authentication	1	
Command Processor	2*	

* If syntax required, 6 commands

Basis of estimate

TBD

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation	U	3
Users Guide	U	18*
API Interface Document	U	5
Interface Design Document		
Test Procedure	U	10

* If syntax required, 6 commands

Assumptions

None.

Open Issues

Provide a list of open issues if there are any. If there are none state none.

COTS Product Dependency List

None.

2.4 SYSTEM VIEWERS

System Viewers Work Required

1. Update System Status Viewer to include new SCT information
2. Provide display of System Performance Data
3. *Start and Stop Subsystem Tests by feature and feature group*

COTS products will be surveyed to determine if the performance data requirements can be met with currently available software products.

CSCI Assessment

The labor costs in the table below are for the labor to produce the product in Atlas.

CSC Name	CSC Labor (LM)	% of CSC
System Status Viewer	4	
Performance/Capacity Monitor	4	

Basis of estimate

TBD

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation	U	
Users Guide	U	
API Interface Document		
Interface Design Document		
Test Procedure	U	

Assumptions

None.

Open Issues

None.

COTS Product Dependency List

TBD.

2.5 SYSTEM SERVICES ASSESSMENT

Network Services Work Required

Reliable Messages must provide interfaces to allow System Integrity to provide notification of subsystem switch-over, command network to dual or single network mode, and select the active network. Reliable Messages must input a health counter to subsystem integrity, notify subsystem integrity of errors, and report performance parameters to subsystem integrity. Reliable messages must provide a high priority message capability to allow System Integrity to issue system event codes in a timely manner.

Reliable Messages Work Required

CSCI Assessment

CSC Name	CSC Labor (LM)	% of CSC
Reliable Services	*	

Basis of estimate

Within scope of existing work.

Documentation

Assumptions

Open Issues

COTS Product Dependency List

Initialization and Termination Services Work Required

Initialization and Termination Services contains functions that duplicate subsystem integrity functions. These include:

1. Application Registration
2. Restart Application
3. Alive Test
4. Kill process

Each of these functions must be reviewed to determine the appropriate implementation which may include any of the following:
1. Move the code to System Integrity
2. Rewrite function in System Integrity
3. Provide an API for System Integrity to use existing ITS function

CSCI Assessment

The labor costs in the table below are for the labor to produce the product in Atlas.

CSC Name	CSC Labor (LM)	% of CSC

Basis of estimate

TBD

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation	U	
Users Guide	U	
API Interface Document		
Interface Design Document		
Test Procedure	U	

Assumptions

None.

Open Issues

COTS Product Dependency List

None.

2.6 DATA DISTRIBUTION ASSESSMENT

Data Distribution Work Required

The following work items are required in Data Distribution

1. Incorporate Health Counter API calls
2. *Accept system control command to assume active/standby role*
3. *Provide CVT updates from standby when it becomes active*
4. *Develop method to synchronize Data Fusion after switch-over*
5. *Develop method to resume Constraint Management processing when standby DDP becomes active*
6. Send Packet missed message to Subsystem Integrity when gateway fails to send expected change data packet (both active and standby)
7. Request data from standby gateway when packet not received from active gateway.
8. Change mechanism for setting data health bad on failure of a primary.
9. Define Pseudo FDs processing after switch-over.
10. *Determine the method for inhibiting the output of change data, data fusion, and constraint management from standby DDP*

Italicized entries under review. New assessments required.

CSCI Assessment

CSC Name	CSC Labor (LM)	SLOCs
Data Distribution	15 LM	3,000
Data Fusion	1 LM	100
Constraint Management	3 LM	700

Basis of estimate

TBD

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation	Update	TBD
Users Guide	N/A	N/A
API Interface Document	N/A	N/A
Interface Design Document	N/A	N/A
Test Procedure	Update	TBD

Assumptions

Open Issues

- How to unit test at SDE-Houston

TRAVEL REQUIREMENTS

From	To	Reason	No. People	Duration	Est. Date or Frequency
Houston	KSC	Support of Design Panels by Houston Developers/management	3	1 week per trip	3 trips
Houston	KSC	On-site integration testing and CIT	3	3 weeks per trip	3 trips
Houston	KSC	On-site integration support	2	2 weeks per trip	3 trips
KSC	Houston	Design coordination	2	1 week per trip	2 trips

SCHEDULES

DP2 – 4/21 internal
4/23 final

DP3 - 5/19 internal
5/21 final

CSC Name	CSC Labor (LM)	% of CSC

Basis of estimate

TBD

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation		
Users Guide		
API Interface Document		
Interface Design Document		
Test Procedure		

Assumptions

Open Issues

COTS Product Dependency List

Product Name	Quantity Needed	Need Date

2.7 LDB GATEWAY ASSESSMENT

LDB Work Required

TBD.

CSCI Assessment

The labor costs in the table below are for the labor to produce the product in the Atlas release.

CSC Name	CSC Labor (LM)	% of CSC

Basis of estimate

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation		
Users Guide		
API Interface Document		
Interface Design Document		
Test Procedure		

Assumptions

Open Issues

The interface between LDB and System Integrity needs to be fully defined, including:
- Error detection and reporting cases for LDB, i.e. what errors are detected by LDB and how they are reported.
- System Event Codes associated with switch-over, both sent and received by LDB
- Commands associated with switch-over.

COTS Product Dependency List

2.8 PCM DOWNLINK GATEWAY

Add the capability to checksum each frame of the input PCM Telemetry stream and output the checksum along with the frame counter each SSR.

CSCI Assessment

The labor costs in the table below are for the labor to produce the product in the Atlas release.

CSC Name	CSC Labor (LM)	% of CSC

Basis of estimate

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation		
Users Guide		
API Interface Document		
Interface Design Document		
Test Procedure		

Assumptions

None.

Open Issues

The interface between PCM Downlink gateway and System Integrity needs to be fully defined, including:

- Error detection and reporting cases for PCM, i.e. what errors are detected by PCM D/L and how they are reported.
- System Event Codes associated with switch-over, both sent and received by PCM D/L
- Commands associated with switch-over.

COTS Product Dependency List

None.

2.9 GSE GATEWAY ASSESSMENT

GSE GATEWAY SERVICES ASSESSMENT

This CSCI must be modified to incorporate Active / Standby functionality including error detection and switch-over.

GSE Gateway Command Processor CSC Work Required

The GSE Gateway Command Processor CSC must be modified to provide the command functionality required of a standby GSE gateway. This includes monitoring the active GSE gateway bus commands and HIM responses. Any

RTCN HIM commands which were received by the standby and were not sent to the HIM by the active prior to a switch-over command will be sent by the standby (now active) after the switch-over.

Initialization CSC Work Required

The Initialization CSC must be modified to allow gateway activation in standby mode

Issue Command CSC Work Required

The Issue Command CSC must be modified to provide the standby functionality of capturing the active gateway's HIM command/measurement requests and their associated HIM response and forwarding this information to the proper CSC (either Command Processor or Measurement Processing)

Measurement Processing CSC Work Required

The Measurement Processing CSC must be modified to provide the measurement poll functionality required of a standby GSE gateway. This includes monitoring the active GSE gateway measurement polls and HIM responses. The standby gateway will track the active through the poll tables and verify the active gateway's poll cycle is correct. When a switchover is commanded, the standby (now active) will pick up in the poll table where the active left off.

Subsystem Integrity CSC Work Required

This CSC is new for Atlas and is the major CSC associated with active/standby and subsystem integrity. This CSC will be responsible for all commands, health/status monitoring and system event code generation and receipt associated with subsystem integrity.

CSCI Assessment

CSC Name	CSC Labor (LM)	% of CSC
Command Processor	0.5	100
Initialization	0.5	100
Issue Command	0.5	100
Measurement Processing	2	100
Subsystem Integrity	1	80

Basis of estimate

CSC Name	LOC
Command Processor	200
Initialization	100
Issue Command	100
Measurement Processing	300
Subsystem Integrity	200

Documentation

Document Type	New/Update	Number of Pages
Requirements and Design Documentation	Update	20
Users Guide	Update	5
API Interface Document	Update	10
Interface Design Document		
Test Procedure	Update	20

Assumptions

None

Open Issues

The interface between GSE and System Integrity needs to be fully defined, including:

- Error detection and reporting cases for GSE, i.e. what errors are detected by GSE and how they are reported.
- System Event Codes associated with switch-over, both sent and received by GSE
- Commands associated with switch-over.

COTS Product Dependency List

None

3. HWCI ASSESSMENTS

Not Applicable.

APPENDIX A
Definitions

Availability — The percentage of the time a system, component or application is available to the user. Availability can be calculated with the equation:

A = MTBF/(MTBF + MTTR)

Bayesian Method — A technique for managing uncertainty based on event probability. See Bayes' Rule.

Bayes Rule — The probability of event *y* occurring given that event *x* is true is $p(x \mid y)*p(y)/p(x)$. Where $p(x \mid y)$ is the probability of *x* occurring given *y* is true.

Critical Process — A software or hardware process in a subsystem that must be operating properly for the subsystem to be in the "GO Mode"

Failure — 1) the inability of a system, subsystem, component, or part to perform its required function within specified limits, under specified conditions for a specific duration. 2) Loss of proper service suffered by the user. Denotes an element's inability to perform its designed function because of an error(s) in the element or its environment, which, in turn, is caused by a fault(s). The result of both a fault and a stimulus, i.e. a condition that caused the fault to be uncovered, resulting in a deviation from specified output. Examples of the stimulus could be input data and the computer data (exception conditions arising form overload or hardware anomalies).

Failure Mode and Effects Analysis (FMEA) — Study of a system and the working interrelationships of its elements to determine ways in which failure can occur (failure modes) and the effects of each potential failure on the system element in which it occurs, on other system elements, and on the success of the system's mission.

Fault — 1) A condition that may cause a functional unit to fail to perform its required function. 2) An anomalous physical condition, defect or bug in the delivered software or hardware which has the potential to cause errors and failures. In software a fault is a design/implementation flaw. Hardware faults include design, implementation, or manufacturing flaws and intrinsic failures.

Fault Detection. — A process that discovers or is designed to discover faults; the process of determining that a fault has occurred.

Fault Isolation. — The process of determining the location or source of a fault.

Fault Recovery. — A process of elimination of a fault without permanent reconfiguration.

Fault Tolerance — Fault tolerance is defined as the property of a system "to provide, by redundancy or high reliability, service complying with the specification in spite of faults having occurred or occurring".

Reliability — The probability that a system, component or application will perform its specified function for a specified time under specified conditions.

Non Critical Process — A software or hardware process in a subsystem that is not required to be operating properly for the subsystem to be in the "GO Mode"

Subsystem — The collection of hardware and software that is combined to perform a specific set of functions (e.g. GSE Gateway, CCP, DDP). A subsystem in constrained to a single computer platform.

Subsystem Health FD — The single FD from each subsystem that indicates the subsystem is performing its function.

Subsystem Status FDs — The set of FDs that provide detailed status of the subsystem (e.g., Use & Error Counters, Format IDs being processed, etc.)

System Failure —

Full Capability Mode —

Reduced Capability Mode —

Emergency Safing Mode —

APPENDIX B

Example Rules For GS1A Failure

GS1A reports missing HIM responses 1	GS1S reports missing HIM responses 2	GS1S reports missing GS1A polls 3	DDPA reports missing GS1A packet 4	DDPS reports missing GS1A packet 5	DDPA reports missing GS1S requested packet 6	DDPS reports missing GS1S requested packet 7

ASSUMPTION - GS1A, GS1S, DDPA, DDPS ARE IN GO MODE. (NEED ANOTHER TRUTH TABLE FOR OTHER CONFIGURATIONS)

1	2	3	4	5	6	7	MEANING
						Y	DDPS is down - It missed a GS1S packet but DDPA did not
					Y		DDPA is down - It missed a GS1S packet but DDPS did not
					Y	Y	GS1S is down - Both DDPs missed a GS1S packet
				Y			DDPS is down - It missed a GS1A packet but DDPA did not
				Y		Y	DDPS is down - It missed a GS1A packet and a GS1S packet but DDPA did not
				Y	Y		
				Y	Y	Y	
			Y				DDPA is down - It missed a GS1A packet but DDPS did not
			Y			Y	
			Y		Y		DDPA is down - It missed a GS1A packet and a GS1S packet but DDPS did not
			Y		Y	Y	
			Y	Y			GS1A is down - Both DDPs missed a GS1A packet
			Y	Y		Y	
			Y	Y	Y		
			Y	Y	Y	Y	**CALL GOD**
		Y					GS1S is down - It reported missing GS1A polls but DDPs did not miss GS1A packets. GS1S has bad RX/TX
		Y				Y	
		Y			Y		
		Y			Y	Y	
		Y		Y			
		Y		Y		Y	
		Y		Y	Y		
		Y		Y	Y	Y	
		Y					
		Y				Y	
		Y			Y		
		Y			Y	Y	
		Y		Y			
		Y		Y		Y	
		Y		Y	Y		
		Y		Y	Y	Y	
		Y	Y				
		Y	Y			Y	
		Y	Y		Y		
		Y	Y		Y	Y	
		Y	Y	Y			GS1A is down - Both DDPs missed GS1A packets and GS1S missed GS1A polls
		Y	Y	Y		Y	

1	2	3	4	5	6	7	MEANING
		Y	Y	Y	Y		
		Y	Y	Y	Y	Y	
	Y						GS1S is down - It did not see HIM responses but GS1A did
	Y					Y	
	Y				Y		
	Y				Y	Y	
	Y			Y			
	Y			Y		Y	
	Y			Y	Y		
	Y			Y	Y	Y	
	Y		Y				
	Y		Y			Y	
	Y		Y		Y		
	Y		Y		Y	Y	
	Y		Y	Y			GS1A is down - GS1S did not see HIM responses and both DDPs missed GS1A packet. GS1A has bad path between RX/TX and XMITTER
	Y		Y	Y		Y	
	Y		Y	Y	Y		
	Y		Y	Y	Y	Y	
	Y	Y					GS1A is down - GS1S does not see polls or HIM responses
	Y	Y				Y	
	Y	Y			Y		
	Y	Y			Y	Y	
	Y	Y		Y			
	Y	Y		Y		Y	
	Y	Y		Y	Y		
	Y	Y		Y	Y	Y	
	Y	Y					
	Y	Y				Y	
	Y	Y			Y		
	Y	Y			Y	Y	
	Y	Y		Y			
	Y	Y		Y		Y	
	Y	Y		Y	Y		
	Y	Y		Y	Y	Y	
	Y	Y	Y				
	Y	Y	Y			Y	
	Y	Y	Y		Y		
	Y	Y	Y		Y	Y	
	Y	Y	Y	Y			GS1A is down - GS1S does not see polls or HIM responses and DDPs missed GS1A packet
	Y	Y	Y	Y		Y	
	Y	Y	Y	Y	Y		
	Y	Y	Y	Y	Y	Y	
Y							GS1A is down - It does not see HIM responses but GS1S does. GS1A has bad RX
Y						Y	
Y					Y		
Y					Y	Y	
Y				Y			
Y				Y		Y	
Y				Y	Y		

1	2	3	4	5	6	7	MEANING
Y				Y	Y	Y	
Y			Y				
Y			Y			Y	
Y			Y		Y		
Y			Y		Y	Y	
Y			Y	Y			
Y			Y	Y		Y	
Y			Y	Y	Y		
Y			Y	Y	Y	Y	
Y		Y					GS1A is down
Y		Y				Y	
Y		Y			Y		
Y		Y			Y	Y	
Y		Y		Y			
Y		Y		Y		Y	
Y		Y		Y	Y		
Y		Y		Y	Y	Y	
Y		Y					
Y		Y				Y	
Y		Y			Y		
Y		Y			Y	Y	
Y		Y		Y			
Y		Y		Y		Y	
Y		Y		Y	Y		
Y		Y		Y	Y	Y	
Y		Y	Y				
Y		Y	Y			Y	
Y		Y	Y		Y		
Y		Y	Y		Y	Y	
Y		Y	Y	Y			
Y		Y	Y	Y		Y	
Y		Y	Y	Y	Y		
Y		Y	Y	Y	Y	Y	
Y	Y						BUS 1 is down
Y	Y					Y	
Y	Y				Y		
Y	Y				Y	Y	
Y	Y			Y			
Y	Y			Y		Y	
Y	Y			Y	Y		
Y	Y			Y	Y	Y	
Y	Y		Y				
Y	Y		Y			Y	
Y	Y		Y		Y		
Y	Y		Y		Y	Y	
Y	Y		Y	Y			
Y	Y		Y	Y		Y	
Y	Y		Y	Y	Y		
Y	Y		Y	Y	Y	Y	BUS 1 is down
Y	Y	Y					GS1A is down
Y	Y	Y				Y	

1	2	3	4	5	6	7	MEANING
Y	Y	Y			Y		
Y	Y	Y			Y	Y	
Y	Y	Y		Y			
Y	Y	Y		Y		Y	
Y	Y	Y		Y	Y		
Y	Y	Y		Y	Y	Y	
Y	Y	Y					
Y	Y	Y				Y	
Y	Y	Y			Y		
Y	Y	Y			Y	Y	
Y	Y	Y		Y			
Y	Y	Y		Y		Y	
Y	Y	Y		Y	Y		
Y	Y	Y		Y	Y	Y	
Y	Y	Y	Y				
Y	Y	Y	Y			Y	
Y	Y	Y	Y		Y		
Y	Y	Y	Y		Y	Y	
Y	Y	Y	Y	Y			GS1A is down
Y	Y	Y	Y	Y		Y	
Y	Y	Y	Y	Y	Y		
Y	Y	Y	Y	Y	Y	Y	

APPENDIX C

HEALTH COUNTERS

Periodic Processes Health Counter

Each subsystem process must include a heartbeat which is invoked periodically to let SSI know that the process is still cycling. This heartbeat requires that 2 lines be added to each process. The first is part of the process initialization and creates the heartbeat instance. This initialization statement gives SSI information that allows it to match up the process with expected processes, and tells SI how often the process should be expected to cycle. The second statement actually generates the heartbeat. The heartbeat generation increments a shared memory value and consumes minimal resources. It is not necessary, nor is it acceptable to put the heartbeat in a separate process. It does not consume enough resources to be noticeable, and putting it in a separate process defeats the purpose - to ensure that the process is cycling and not blocked, deadlocked, or in an infinite loop. An example is shown below.

```
// Initialization Subsection
// declaration of instance Heart
// Name should match that provided by ITS,
// Period in milliseconds (example is 1 second)
// Period of MAXINT is assumed to be truly aperiodic, with no guaranteed cycle rate.
// True = cyclic, False = acyclic with maximum time period between cycles of Period
SSIProcessHeart  Heart ("Process Name", 1000, True);          // ← Required Statement

// Realtime Loop subsection
while (Continue) {
        //Wait on some event
        tbd.statements("receive some event");

        Heart.beat ();                                        // ←Required Statement

        //process the event
        tbd.statements();
}
```

Proposal for Health Counter for Non-Periodic Processes

The following discussion is one approach to providing a health count mechanism for non-periodic critical software functions. The Health Counter of a non-periodic software function is a set of two counters, one incremented on entry and one incremented on exit. If the two counters are equal, the function is not active. If the entry counter is 1 greater than the exit counter, the function is in progress. If the two counters remain unequal at the same values for more than a specified maximum amount of time, the software is in an infinite loop. If the exit counter is greater than the entry counter (except for wraparound) a fault has occurred.

APPENDIX D
REDUNDANCY MANAGEMENT SLS REQUIREMENT STATUS

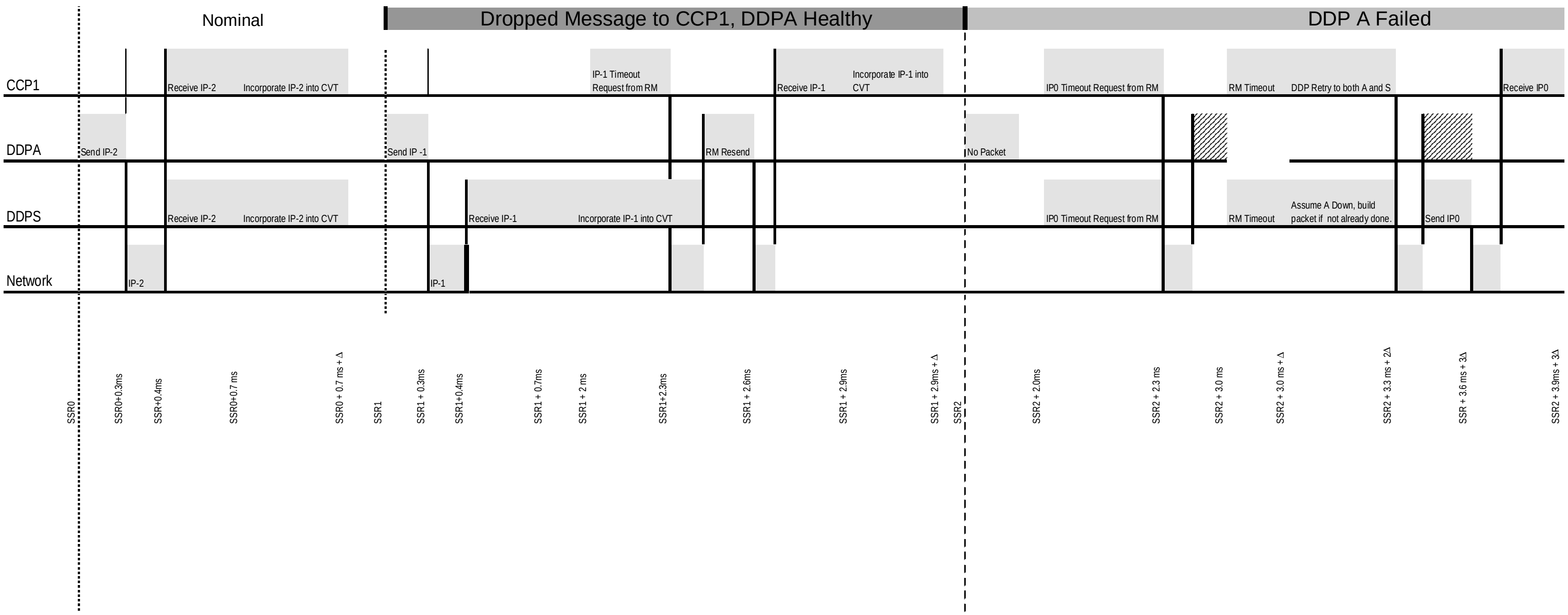
SLS REQ.	Description	Del. Doc. State	Projected State
2.1.1.2.12	The RTPS shall provide the capability to perform continuous fault detection and isolation of RTPS subsystem's hardware.	High	
2.1.1.2.13	*The RTPS shall provide a set of non-intrusive test programs to test interfaces and subsystem LRUs in the RTPS.*	*Low*	
2.2.9.1.5	The RTPS shall provide a set of visual displays that provide comprehensive insight into the state and configuration of the set resources (e.g., *network resources*, subsystem assignments, software configuration, etc.).	High	
2.2.9.1.6	The RTPS shall provide different views of Test Sets and activities in configurable sets (e.g., *Master Set View*, Test Set View, Activity View).	High	
2.2.9.2.8	The RTPS shall provide a visual display depicting the health and status of all hardware resources within a Test Set and *within all Test Sets of a Configurable Set*.	High	
2.2.9.2.9	The RTPS shall provide the capability to monitor the configuration of each subsystem participating in a test including what software is executing and any subsystem error conditions.	Medium	
2.2.9.2.10	The RTPS shall provide a central point for the display of system error, status, and mode change messages.	High	
2.2.9.2.15	The RTPS shall provide the capability to continuously monitor subsystem resource utilization in all RTPS subsystems.	Medium	
2.2.9.3.1	The RTPS shall provide redundancy management of all redundant subsystems and network resources in a Test Set.	Medium	
2.2.9.3.2	The RTPS shall provide a central point to coordinate and direct redundant element activation (known as System Integrity).	Complete	
2.2.9.3.3	System Integrity shall be capable of being run from any Console Position within a Test Set	Complete	
2.2.9.3.4	*The RTPS shall provide the capability for a Standby copy of System Integrity to run and monitor the activities of the Active copy of System Integrity.*	*(Titan) [Reference]*	
2.2.9.3.5	*When the Standby copy of System Integrity determines that the Active copy is not operating properly it shall assume the role and responsibilities of the Active System Integrity.*	*(Titan) [Reference]*	
2.2.9.3.6	The RTPS shall provide a method to share current configuration data with a redundant element.	Complete	
2.2.9.3.7	The RTPS shall provide a method to track redundant element states.	Complete	
2.2.9.3.8	System Integrity shall monitor critical subsystems for failure and in the event a monitored subsystem fails, *shall perform a switch-over* (if enabled) to the standby subsystem.	Medium	
2.2.9.3.9	System Integrity shall report all subsystem errors to a central point	Complete	
2.2.9.3.10	*The CLCS shall provide a reduced capability mode in which a Test Set continues to support even though all copies of System Integrity fail.* **Note:** Functions that are not supported or whose capability is reduced in the reduced capability mode are: 1. Redundant Element Switchover 2. Test Set Resource Monitoring 3. Checkpointing 4. Restarting subsystems	*(Titan) [Partial]*	
2.2.9.3.12	*The CLCS shall provide a "warm boot" capability in which System Integrity can be restored after failure.*		

2.2.9.3.13	*After a "warm boot" System Integrity shall restore normal function to those capabilities which were reduced while in the reduced capability mode.*		
2.2.2.3.1	GSE and PCM Gateways, configured as a redundant pair, shall switch to the standby Gateway with no loss of measurement data and within 1 System Synchronous Rate Time Period of detection.		
2.2.2.3.2	For Gateways (except LDB) configured as a redundant pair, switch-over for commands shall be completed in less than 20 milliseconds without any loss of commands.		
2.2.2.3.3	LDB Gateway switch-over shall be accomplished without any loss of data or commands and shall be completed in less than 500 milliseconds.		
2.2.1.1.1	The RTPS shall be designed to be Fail Safe.	Partial	
2.2.1.1.2	• The RTPS shall be fault tolerant. Specifically, the system shall provide the capability to recover from subsystem failures in the following areas: • Command and Control Processing • Data Distribution Processing [Partial] • Critical Data Acquisition Gateways (i.e., LDB, 128 & 192 Kb PCM, GSE) [Partial] • Real Time Critical Network and the Display and Control Network	Partial Partial Partial Complete	
2.2.1.1.3	• The CLCS shall be designed to have a high level of data integrity. Specifically the system shall provide the following: • No loss of command data within the CLCS • No loss of measurement data within the CLCS • No loss of measurement samples to applications requesting such service • No data which has been corrupted within the CLCS • Health data on a measurement basis	Partial Partial Partial Partial Partial	
2.2.1.1.4	The RTPS shall provide fault tolerance in the Command and Control HCI positions	Complete	
2.2.1.1.5	The loss of any RTPS Real Time Network component shall not cause switch-over of more than one standby subsystem	Complete	
2.2.9.3.9	System Integrity shall report all subsystem errors to a central point.	Complete	
2.2.9.3.8	System Integrity shall monitor critical subsystems for failure and *in the event a monitored subsystem fails, shall perform a switch-over (if enabled) to the standby subsystem.*	Partial	

APPENDIX E
RTPS System States

SYSTEM STATES	VISIBLE TO TC	SYSTEM CAPABILITIES IN THIS STATE	ACCEPTABLE COMMANDS	CMD CAUSING TRANSITION NEXT STATE	RESULT OF COMMAND	RETURN TO STATE COMMAND	CCMS STATE
In Config	X	None	Power On				Power Off
				Power On Boot	OS Loaded & Initialized (GW Init SCID)		
Platform Init'd (GW - N/A)		UNIX Based Comm	Init SCID			Power Off	
				Init SCID	SCID Initialized & Limited Comm		
SCID Initialized		RTPS Comm (Limited)	Init SCID or TCID			Shutdown	Boot Fill
				Init TCID	TCID Initialized		
Loaded	X	RTPS Comm (Limited)	Activate Cmd			Init SCID, Init TCID, Shutdown	
				Activate	HC Started & Full Comm		
Comm	X	Health Counts, Full Comm	Nearly Full Comm, No End Item EI Cmds, MDTM			Terminate, Shutdown	
				A DA	Starts Processing, Data Acquisition		
GO Operational (Active)	X	Data Acq, Full Comm, SSI Poll &Xmit Data Changes.	All except Init SCID, TCID, Activate			Terminate, Shutdown, I DA	A DA
GO Operational (Standby)	X	Data Acq, Full Comm, SSI, Monitor Poll, No Xmit	All except Init SCID, TCID, Activate			Terminate, Shutdown, I DA	A DA
Subsystem GO	X	Application Execution CCP, DDP, CCWS only	All		Application Software Manager Started	Terminate, Shutdown, I DA	
ORT	X	Full comm., Diagnostics			Begins Operational Readiness Test	Exit ORT, Terminate, Shutdown	ORT

APPENDIX F
Failover Time Line



THIS IS THE LAST PAGE OF THE DOCUMENT